Database Systems

Lecture #1

Dr Sherin ElGokhy

References

1. C. J. Date, “Introduction to Database Systems”, Addison Wisley, 7th edition, 2000.
2. R. Elmasry and S. B. Navathe, “Fundamentals of Database Systems”, Addison Wisley, 3th edition, 2000.
3. P. Atzeni, S. Ceri, S. Paraboschi and R. Torlone, “Database Systems: Concepts, Languages, and Architecutres”, Mc. GrawHill, 1999.
4. M. Gruber, “Mastering SQL”, Sybex Inc., 2000

Intended Learning Outcomes ILOs

- Studying the basic concepts of Database: Database Management Systems (DBMS)
- Studying Data Models especially Relational Model
- Studying the Structured Query Language (SQL)

Chapter 1: Introduction

Computer-based Systems

Collection, organization and storage of data are major tasks in many human activities and in every computer-based system.

Examples of types of data

- Bank balances
- Telephone numbers in directories
- Stock quotations
- Availability of credit card funds
- Registers of students enrolled in a university and grades in their exams.

Computer-based systems ensure that this data is permanently maintained, quickly updated to show changes, and made accessible to the queries of users who can be remote from one another and from the computer where the data is kept.

For example, queries about the availability of funds on credit cards, asked through simple devices available in millions of businesses (whether hotels, shops or companies), which allow purchases made anywhere in the world to be charged to the credit card owners.

Information system

- Every organization has an information system, which manages (gets, processes, stores, communicates) the information necessary to perform the functions of the organization.
- The information system possibly not made explicit in its structure.
- Usually, the information system operates in support to other components of the organization.
- The very notion of information system is partly independent of its computerization; however, we are mainly interested in information systems that are, to a large extent, computerized.

Management of information

- Information is handled and recorded according to various techniques:
  - informal ideas
  - natural language (written or spoken)
  - drawings
  - diagrams
  - numbers
  - codes

Information and data

- In most computer-based systems **information is represented by means of data,** which needs to be interpreted in order to provide information.
  - **data** raw facts, alone has no significance, but once interpreted and suitably correlated, it provides **information** that allows us to improve our knowledge of the world.

An example:

– "John Smith" and 25755 are a name (or, better, a string) and a number: two pieces of data

– if they are provided as a reply to a request: "Who is the dept. head, and which is his/her phone number," then we get information out of them

Database

(generic definition)

- A collection of data, used to represent information of interest to an information system.

(more technical definition)

- A collection of data, managed by DBMS.

Database Management System-DBMS

- A database management system (DBMS) as a software system able to manage collections of data that are ***large, shared*** and ***persistent***, and to ensure their **reliability** and ***privacy***. Like any software product, a DBMS must be ***efficient*** and ***effective***.

- A database is a collection of data managed by a DBMS.

Database Management System-DBMS

Is a software system able to manage collections of data that are :

large (bigger, often much bigger, than the main memory available. As a result, a DBMS must manage data in secondary memory)

shared (used by various applications and users, in this way the redundancy of data is reduced, since repetitions are avoided, and, consequently, the possibility of inconsistencies is reduced- In order to guarantee shared access to data by many users operating simultaneously, the DBMS makes use of a special mechanism called concurrency control)

persistent (with a lifespan that is not limited to single executions of the programs that use them)

Database Management System-DBMS

And to ensure their:

reliability (so preserving the database case of hardware or software failure- To fulfill this requirement, DBMSs provide specific functions for backup and recovery) **and**

privacy (controlling accesses and authorizations).

Like any software product, a DBMS must be

efficient (using the capacity to carry out operations using an appropriate amount of resources (time and space) for each user) **and**

effective (supporting the productivity of its users).

Sharing

- Most organizations have a structure (departments, divisions, …) and each component is interested in a portion of the information system
- The data of interest of the various components often overlap
- A database is an **integrated resource**, shared by various components
- Integration and sharing allow a reduction of **redundancy** and the consequent possibility of **inconsistency**
- Since sharing is never complete, DBMS provide support for privacy of data and access authorizations
- Sharing also requires that multiple accesses to data are suitably organized: **concurrency control** techniques are used

DBMS vs file system

- Traditionally, the management of large and persistent sets of data can be done by means of simpler tools: file systems- individual, unrelated files.
- Comparing DBMS vs. file system, several advantages for a DBMS can be mentioned.

1. **Less redundancy**

   In a file system there is a lot of redundancy.

2. **Inconsistency avoidance**

   If the same piece of information is stored in more than one place, then any changes in the data need to occur in all places that data is stored.

DBMS vs file system

3. **Efficiency**

   A DBMS is usually more efficient (time and space) than a file system, because a piece of information is stored in fewer locations

4. **Data integrity**

   In a DBMS, it is easier to maintain data integrity.

- File systems provide also rough support for sharing
- There is no sharp line between DBMSs and non-DBMSs: DBMSs provide many features, that extend those of file systems
- The crucial issue is **effectiveness**, take advantage of these features

DBMS vs file system (2)

- In traditional programs that make use of files, each program includes a description of the organization of the file, which is often just a stream of bytes; there are chances of incoherence between the file and its description (or descriptions, if the file is shared)

- In DBMSs, there is a portion of the database (called the **dictionary** or **catalogue**) that describes the database itself, which is shared

Data Models

Data Model : is a set of constructs used to organize data.

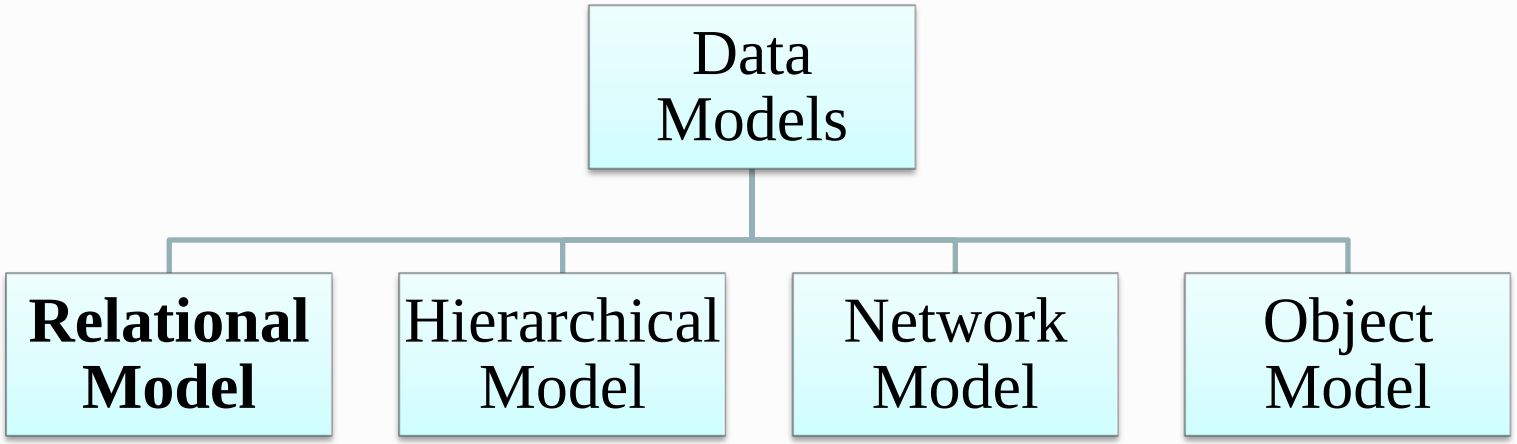


Table: Relation
Row: Tuple or record
The most widespread

Tree

Graph
CODASYL model
Conference on Data Systems Languages
that gave it a precise definition

objects

(**These data models are called "Logical Models".**

Two main types of models

Models

Logical models

- used in DBMSs for the organization of data at a level that although abstract from physical structures, it reflects a particular organization
- examples: **relational, network, hierarchical, object**

Conceptual models

- used to describe data in a way that is completely independent of any system, with the goal of representing the concepts of the real world rather than the data needed for their representation; they are used in the early stages of database design
- the most popular is the **Entity-Relationship(E-R)** model

Relational Model

Example:

TEACHING

Course	Tutor
Database	Samir
Network	Nagy
Languages	Morad

PROSPECTUS

DegreeProgram	Subject	Year
Information Systems	Database	4
Information Systems	Network	4
Information Systems	Languages	3
Electrical Engineering	Database	4
Electrical Engineering	Networks	4

Relational database having two relations:
TEACHING and PROSPECTUS

Schemas and Instances

The **schema** in the database: - is a part that invariant in time (stable over time)

Attributes - made up of the characteristics of data

→ **Intensional Component**

The **instance** in the database: - is a part that can change with time

Rows - made up of the actual data

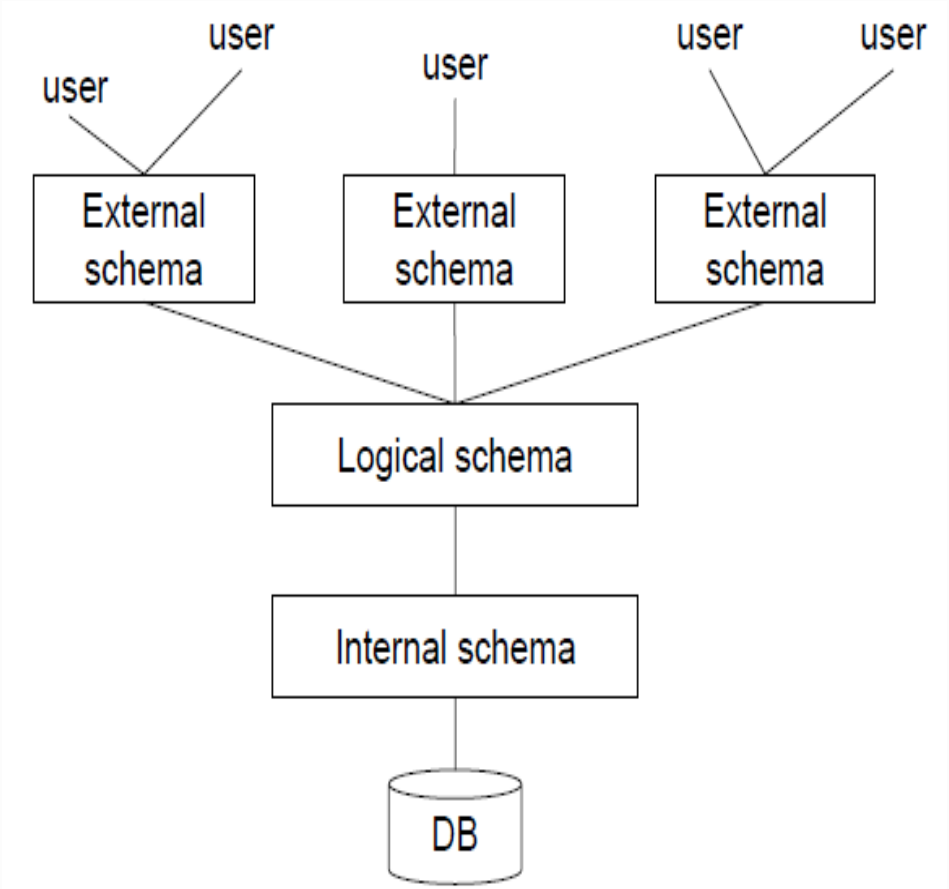
→ **Extensional Component**

Example:

Schema : TEACHING(Course, Tutor)

Instance: Database, Samir, Information System, 4

Abstraction levels in DBMS



External level

Logical level

Internal level

Three level (Schema) Architecture

Abstraction levels in DBMS

1. **Logical schema** is a description of a **whole** database by means of the logical model used by the DBMS (relational, hierarchical, network, object)
2. **Internal schema** describes the implementation of the logical schema by means of the physical **storage** structure:
   - Sequential file
   - Hash file
   - Sequential files with indices
3. **External schema** is the description of a **portion** of the database by means of the logical model. An external schema can offer a different organization of the data in order to reflect the point of view of a particular user or group of users- VIEWS.

   Mechanisms for access authorization can be associated with external schemas, in order to regulate the access of users to the database: a user could be authorized to manipulate only the data described by means of his external schema.

Data independence

- ***Physical independence***: the logical and external level are independent of the physical one;
  - a relation (the high level description) is not influenced byits physical implementation (which could even vary over time)
- ***Logical independence***: the external level is independent of the logical one
  - addition of (or changes to) views do not require changes to the logical schema
  - changes to the logical schema need not affect the external schemas (provided that the definition of mappings are adjusted)

Database languages

- Various forms (a contribution to effectiveness)
  1. Interactive textual languages, such as SQL
  2. Interactive commands **embedded** in a **host** language (Pascal, C, Cobol, Java, etc.)
  3. Interactive commands **embedded** in an **ad-hoc** development language, usually with additional features (for the production of forms, menus, reports, ...)
  4. By means of non-textual **user-friendly** interfaces- allow the formulation of queries without the use of a textual language

- ✓ SQL
- ✓ SQL embedded in traditional programming languages (Pascal, C++, ...)
- ✓ SQL embedded in ad-hoc languages (Microsoft Access, Oracle, …)

Database languages

✓SQL

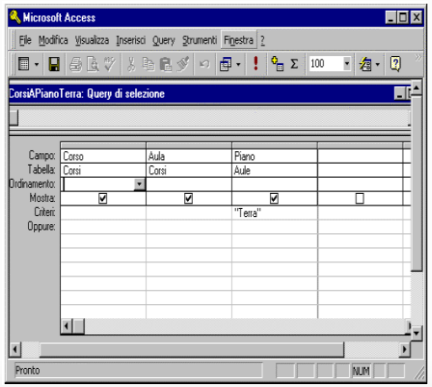
```
SELECT Course, Room, Floor
FROM Rooms, Courses
WHERE Code = Room
AND Floor="Ground"
```

Course	Room	Floor
Networks	N3	Ground
Systems	N3	Ground

✓SQL embedded in traditional programming languages (Pascal, C++, ...)

```
write('city name"?'); readln(city);
EXEC SQL DECLARE E CURSOR FOR
      SELECT NAME, SALARY
      FROM EMPLOYEES
      WHERE CITY = :city ;
EXEC SQL OPEN E ;
EXEC SQL FETCH E INTO :name, :salary ;
while SQLCODE = 0 do begin
    write(employee:', name, 'raise?');
    readln(raise);
    EXEC SQL UPDATE PERSONE SET SALARY = SALARY + :raise
                WHERE CURRENT OF E
    EXEC SQL FETCH E INTO :name, :salary
  end;
EXEC SQL CLOSE CURSOR E
```

✓SQL embedded in ad-hoc languages (Microsoft Access, Oracle, …)



Database languages

- ***Data Definition Language (DDL):*** used to define the logical, external and physical **schemas** and access authorizations.

- ***Data Manipulation Language (DML):*** used for querying and updating database **instances**

Users and Designers

1. Database Administrator (DBA):
   - is a person responsible for design, control, and administration of database.
2. Application designers and programmers
   - Define and create programs that access the database.
   - Uses DML.
3. Users:
   - End user: predefined activities (transactions)
   - Casual user: specialists in the language (Experience)- able to use the interactive languages to gain access to the database

Advantages and disadvantages of DBMS

- Advantages of DBMS:
  - Common resource.
  - Standard and precise model
  - Centralized control on data
  - Sharing: reduction of redundancy and inconsistency
  - Data independence
- Disadvantages of DBMS:
  - Expensive products, complex and quite different from many other software tools.
  - A whole set of services which necessarily carry a cost can generate inefficiency.

Thanks